

LL		BBBBBBBB	LL	UU	UU	NN	NN
LL		BBBBBBBB	LL	UU	UU	NN	NN
LL		BB BB	LL	UU	UU	NN	NN
LL		BB BB	LL	UU	UU	NN	NN
LL		BB BB	LL	UU	UU	NNNN	NN
LL		BB BB	LL	UU	UU	NNNN	NN
LL		BBBBBBBB	LL	UU	UU	NN NN	NN
LL		BBBBBBBB	LL	UU	UU	NN NN	NN
LL		BB BB	LL	UU	UU	NN NNNN	NN
LL		BB BB	LL	UU	UU	NN NNNN	NN
LL		BB BB	LL	UU	UU	NN NN	NN
LL		BB BB	LL	UU	UU	NN NN	NN
LLLLLLLL		BBBBBBBB	LLLLLLLL	UUUUUUUU	NN	NN
LLLLLLLL		BBBBBBBB	LLLLLLLL	UUUUUUUU	NN	NN

LL		SSSSSSSS
LL		SSSSSSSS
LL		SS
LL		SS
LL		SS
LL		SSSSSS
LL		SSSSSS
LL		SS
LLLLLLLL		SSSSSSSS
LLLLLLLL		SSSSSSSS

(2) 56
(3) 106
(4) 195

DECLARATIONS
LIB\$GET_LUN - Allocate one logical unit number
LIB\$FREE_LUN - Deallocate one logical unit number

0000 1 .TITLE LIB\$LUN - Resource allocator for logical unit numbers
0000 2 .IDENT /1-005/
0000 3 :
0000 4 :
0000 5 *****
0000 6 *
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9 * ALL RIGHTS RESERVED.
0000 10 *
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16 * TRANSFERRED.
0000 17 *
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20 * CORPORATION.
0000 21 *
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24 *
0000 25 *
0000 26 *****
0000 27 :
0000 28 :
0000 29 :++
0000 30 :FACILITY: General Utility Library
0000 31 :
0000 32 :ABSTRACT:
0000 33 :
0000 34 : Two routines for allocating and deallocating logical unit
0000 35 : numbers. Using these routines allows use of logical
0000 36 : unit number by multiple procedures without conflicts.
0000 37 :
0000 38 :ENVIRONMENT: User Mode, AST Reentrant
0000 39 :
0000 40 :--
0000 41 :AUTHOR: R. Reichert, CREATION DATE: 04-JUN-79
0000 42 :
0000 43 :MODIFIED BY:
0000 44 :
0000 45 : 1-001 Original. RKR 04-JUNE-79
0000 46 : 1-002 Removed entry point LIB\$RESERVE_LUN and return status
0000 47 : LIB\$_LUNALRRES (Lun already reserved). RKR 20-JULY-79
0000 48 : 1-003 Comment clean up. Made compares against LOW_LUN and HIGH_LUN
0000 49 : CMPL's. Made all references to LUN_POOL be G^LUN_POOL.
0000 50 : RKR. 24-JULY-79
0000 51 : 1-004 Add a first time flag, and set up initial mask based on that, in order
0000 52 : to allow the linker to perform demand-zero compression. MDL 6-Jul-1984
0000 53 : 1-005 Change interpretation of bits in LUN_POOL to eliminate need for a
0000 54 : first time flag and thus restore AST-reentrancy. MDL 7-Aug-1984

0000 56 .SBTLL DECLARATIONS
0000 57 :
0000 58 : INCLUDE FILES:
0000 59 :
0000 60 :
0000 61 :
0000 62 : EXTERNAL DECLARATIONS:
0000 63 :
0000 64 .DSABL GBL : Prevent undeclared
0000 65 : symbols from being
0000 66 : automatically global.
0000 67 :
0000 68 : Error codes
0000 69 :
0000 70 .EXTRN LIB\$_INSLUN : Insufficient luns
0000 71 .EXTRN LIB\$_LUNALRFRE : Lun already free
0000 72 .EXTRN LIB\$_LUNRESSYS : Lun reserved to system
0000 73 :
0000 74 : MACROS:
0000 75 :
0000 76 : NONE
0000 77 :
0000 78 :
0000 79 : EQUATED SYMBOLS:
0000 80 :
0000 81 : LUN NUMBER = 4 ; logical unit number parameter
0000 82 : HIGH_LUN = 119 ; highest unit number dispensed
0000 83 : LOW_LUN = 100 ; lowest unit number dispensed
0000 84 :
0000 85 : OWN STORAGE:
0000 86 :
0000 87 :
0000 88 .PSECT _LIB\$DATA RD, WRT, NOEXE, NOSHR, PIC, LONG, -
0000 89 CON, LCL, REL, USR
0000 90 :
0000 91 LUN_POOL: ; Pool of available logical unit numbers
0000 92 .LONG 0 :
0004 93 : Low order bit represents logical
0004 94 : unit number 119, high order bit
0004 95 : logical unit number 88.
0004 96 : Only unit numbers 119 to 100 are
0004 97 : dispensed by this routine.
0004 98 :
0004 99 :
0004 100 : PSECT DECLARATIONS:
0004 101 :
0004 102 .PSECT _LIB\$CODE PIC, USR, CON, REL, LCL, SHR, -
0000 103 EXE, RD, NOWRT, LONG
0000 104 :

0000 106 .SBTTL LIB\$GET_LUN - Allocate one logical unit number
0000 107 ++
0000 108 FUNCTIONAL DESCRIPTION:
0000 109
0000 110 LIB\$GET_LUN allocates one logical unit number from a process-wide
0000 111 pool. If a lun is available for use, its number is returned
0000 112 to the caller. If no luns are available, an error is returned
0000 113 as the function value.
0000 114
0000 115 CALLING SEQUENCE:
0000 116
0000 117 status.wlc.v = LIB\$GET_LUN (LUN_NUMBER.wl.r)
0000 118
0000 119 INPUT PARAMETERS:
0000 120
0000 121 NONE
0000 122
0000 123 IMPLICIT INPUTS:
0000 124
0000 125 LUN_POOL, a table of available logical unit numbers in OWN
0000 126 storage.
0000 127
0000 128 OUTPUT PARAMETERS:
0000 129
0000 130 LUN_NUMBER - The number of the logical unit number allocated
0000 131 or -1 if none were available.
0000 132
0000 133 IMPLICIT OUTPUTS:
0000 134
0000 135 If successful, an entry is made into LUN_POOL indicating that
0000 136 a logical unit number has been reserved.
0000 137
0000 138 FUNCTION VALUE:
0000 139 COMPLETION CODES:
0000 140
0000 141 SSS_NORMAL - Routine successfully completed.
0000 142
0000 143 LIB\$_INSLUN - Insufficient logical unit numbers. There were no
0000 144 more logical unit numbers available for allocation.
0000 145 If this error is returned, lun number is
0000 146 also set to -1 in case the caller does not
0000 147 check for failure.
0000 148
0000 149 SIDE EFFECTS:
0000 150
0000 151 NONE
0000 152
0000 153 --
0000 154
4000 0000 155 .ENTRY LIB\$GET_LUN, "M<IV>" : Save nothing
0002 156
0002 157 :+
0002 158 : Scan LUN_POOL for first available logical unit number
0002 159 :-
0002 160
0002 161 SCAN:
0002 162 FFC #0, #20, G^LUN_POOL, R0 : Look at the first 20 bits in LUN_POOL.

15 13 000B 163 : They represent LUNs 119-100.
000B 164 :
000D 165 :
000D 166 :+
000D 167 : Now recheck and set the bit in an uninterruptable fashion,
000D 168 : in case someone has set it at AST level in the meantime.
000D 169 :
000D 170 :
000D 171 FOUND:
ED 00000000'GF 50 E2 000D 172 BBSS R0, G^LUN_POOL, SCAN ; Repeat scan if already set
0015 173 :
0015 174 :+
0015 175 : Return success with logical unit number in lun_number.
0015 176 :
0015 177 :
04 BC 00000077 BF 50 C3 0015 178 SUBL3 R0, #HIGH_LUN, @LUN_NUMBER(AP) ; Subtract from HIGH_LUN
001E 179 : because lo order table bit is
001E 180 : lun 'high_lun'.
001E 181 :
50 01 D0 001E 182 MOVL #1, R0 : SSS_NORMAL
04 0021 183 RET : Exit
0022 184 :
0022 185 :+
0022 186 : Return error since no logical unit numbers available
0022 187 :
0022 188 :
0022 189 ALL_OUT:
50 04 BC 01 CE 0022 190 MNEG L #1, @LUN_NUMBER(AP) : Set LUN_NUMBER to -1
D0 0026 191 MOVL #LIB\$_INSLUN, R0 : Insufficient logical unit numbers
04 002D 192 RET : Exit
002E 193

002E 195 .SBTTL LIB\$FREE_LUN - Deallocate one logical unit number
002E 196 ++
002E 197 FUNCTIONAL DESCRIPTION:
002E 198 LIB\$FREE_LUN is the complement of LIB\$GET_LUN. When a routine
002E 199 called LIB\$GET_LUN to allocate a logical unit number, and no
002E 200 longer needs it, LIB\$FREE_LUN should be called to free the
002E 201 logical unit number for use by other routines.
002E 202
002E 203
002E 204 CALLING SEQUENCE:
002E 205
002E 206 status.wlc.v = LIB\$FREE_LUN (LUN_NUMBER.rl.r)
002E 207
002E 208 INPUT PARAMETERS:
002E 209
002E 210 LUN_NUMBER - The number of the logical unit to be
002E 211 deallocated. This is the value returned
002E 212 to the user by LIB\$GET_LUN.
002E 213
002E 214 IMPLICIT INPUTS:
002E 215
002E 216 LUN_POOL, a table of available logical unit numbers in OWN
002E 217 storage.
002E 218
002E 219 OUTPUT PARAMETERS:
002E 220
002E 221 NONE
002E 222
002E 223 IMPLICIT OUTPUTS:
002E 224
002E 225 An entry is made in LUN_POOL indicating that the logical unit
002E 226 number is free for use.
002E 227
002E 228 FUNCTION VALUE:
002E 229 COMPLETION CODES:
002E 230
002E 231 SSS_NORMAL - Routine successfully completed.
002E 232
002E 233 LIB\$LUNALRFRE - Logical unit number already free.
002E 234
002E 235 LIB\$LUNRESSYS - Logical unit number reserved to system. This
002E 236 occurs if lun_number is outside the range
002E 237 of "LOW_LUN" and "HIGH_LUN".
002E 238
002E 239 SIDE EFFECTS:
002E 240
002E 241 NONE
002E 242
002E 243 --
002E 244
4000 002E 245 .ENTRY LIB\$FREE_LUN, ^M<IV> ; Save nothing
0030 246
0030 247 :+
0030 248 : Check to see if lun_number is in the proper range.
0030 249 :-
0030 250
0030 251 CMPL @LUN_NUMBER(AP), #HIGH_LUN : Bigger than high_lun?

00000077 8F 04 BC D1 0030 251

00000064 BF 04 BC 1F 14 003B 252 BGTR RES_SYS_1 : Yes, error
15 D1 003A 253 CMPL @LUN_NUMBER(AP), #LOW_LUN : Less than lowest ?
19 00/2 254 BLSS RES_SYS_1 : Yes, error
0044 255
0044 256 ;+
0044 257 ;+ LUN_NUMBER is in range. Now, unset the bit.
0044 258 ;-
0044 259
0044 260 OK_1:
50 00000077 BF 04 BC C3 0044 261 SUBL3 @LUN_NUMBER(AP), #HIGH_LUN, R0 : Convert to bit offset
OC 00000000'GF 50 E5 004D 262 BBCC R0, G^LUN_POOL, ALR_FRE : Clear but error if
0055 263 already clear.
0055 264
0055 265 ;+
0055 266 ;+ Return success
0055 267 ;-
50 01 D0 0055 268
04 0058 269 MOVL #1, R0 : SSS_NORMAL
0059 270 RET
0059 271
0059 272 ;+ Error if logical unit number reserved to system or
0059 273 ;+ out of range.
0059 274 ;-
0059 275 ;-
0059 276
50 00000000'8F D0 0059 277 RES_SYS_1:
04 0060 278 MOVL #LIB\$LUNRESSYS, R0 : Logical unit number reserved
0061 279 RET
0061 280
0061 281 ;+
0061 282 ;+ Error if logical unit number already free.
0061 283 ;-
0061 284
50 00000000'8F D0 0061 285 ALR_FRE:
04 0068 286 MOVL #LIB\$LUNALRFRE, R0 : Logical unit number already free
0069 287 RET
0069 288
0069 289 .END

LIBSLUN Symbol table

- Resource allocator for logical unit nu 16-SEP-1984 00:12:59 VAX/VMS Macro V04-00
6-SEP-1984 11:08:55 [LIBRTL.SRC]LIBLUN.MAR;1

Page 7
(4)

ALL_OUT	00000022	R	02
ALR_FRE	00000061	R	02
FOUND	0000000D	R	02
HIGH_LUN	= 00000077		
LIBSFREE_LUN	0000002E	RG	02
LIBSGET_LUN	00000000	RG	02
LIBS_INSLUN	*****	X	00
LIBS_LUNALRFRE	*****	X	00
LIBS_LUNRESSYS	*****	X	00
LOW_LUN	= 00000064		
LUN_NUMBER	= 00000004		
LUN_POOL	00000000	R	01
OK_T	00000044	R	02
RE5_SYS_1	00000059	R	02
SCAN	00000002	R	02

+-----+
! Psect synopsis !
+-----+

PSECT name

<u>Allocation</u>	<u>PSECT No.</u>	<u>Attributes</u>												
00000000	(0.)	00	(0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE
00000004	(4.)	01	(1.)	PIC	USR	CON	REL	LCL	NOSHR	NOEXE	RD	WRT	NOVEC	LONG
00000069	(105.)	02	(2.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG

! Performance indicators !

Phase

Page faults	CPU Time	Elapsed Time
29	00:00:00.04	00:00:00.68
114	00:00:00.32	00:00:02.36
71	00:00:00.38	00:00:02.04
0	00:00:00.00	00:00:00.00
62	00:00:00.36	00:00:02.77
3	00:00:00.01	00:00:00.01
2	00:00:00.01	00:00:00.01
0	00:00:00.00	00:00:00.00
283	00:00:01.12	00:00:07.87

The working set limit was 900 pages.

3030 bytes (6 pages) of virtual memory were used to buffer the intermediate code.

There were 10 pages of symbol table space allocated to hold 15 non-local and 0 local symbols.

289 source lines were read in Pass 1, producing 16 object records in Pass 2.
0 areas of virtual memory were used to define 0 pages.

0 pages of virtual memory were used to define 0 macros.

+-----+
! Macro library statistics !
+-----+

Macro Library name

Macros defined

~~\$255\$DUA28:[SYSLIB]STARLET.MLB:2~~

D

0 GETS were required to define 0 macros.

LIB\$LUN
VAX-11 Macro Run Statistics

M 8
- Resource allocator for logical unit nu 16-SEP-1984 00:12:59 VAX/VMS Macro V04-00
6-SEP-1984 11:08:55 [LIBRTL.SRC]LIBLUN.MAR;1

Page 8
(4)

L1
1-

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:LIBLUN/OBJ=OBJ\$:LIBLUN MSRC\$:LIBLUN/UPDATE=(ENH\$:LIBLUN)

0208 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

